|  |
| --- |
|  |
| Inligtingstegnologie |
|  |
| Praktiese Assesserings taak |
|  |
| Graad 11 |
| 2018 |

|  |
| --- |
|  |

**INHOUD**

[Wat is die PAT? 3](#_Toc511034004)

[Puntetoekenning 3](#_Toc511034005)

[Onderwerp 4](#_Toc511034006)

[Oorsig 5](#_Toc511034007)

[Fase 1 – Analise 5](#_Toc511034008)

[Fase 2– Ontwerp 5](#_Toc511034009)

[Fase 3 – Kodering en Toetsing 5](#_Toc511034010)

[PAT-Vereistes 6](#_Toc511034011)

[woordelys 7](#_Toc511034012)

[Wat jy BEnodig om die PAT te VOLTOOI 8](#_Toc511034013)

[Wangedrag 8](#_Toc511034014)

[Versuim 8](#_Toc511034015)

[Instruksies vir Fase 1 9](#_Toc511034016)

[Instruksies vir Fase 2 11](#_Toc511034017)

[Instruksies vir Fase 3 13](#_Toc511034018)

[Bylaag A – Uitleg van Verslag 23](#_Toc511034019)

[Bylaag B – Egtheidsverklaring 24](#_Toc511034020)

[Bylaag C – Terminologie 25](#_Toc511034021)

[Bylaag D – Voorbeeld 28](#_Toc511034022)

[Riglyne vir onderwysers om die leerders leiding te gee i](#_Toc511034023)

[Wat word daar van die leerders verwag om te doen en te lewer? i](#_Toc511034024)

[Hoe sal die leerders te werk gaan? i](#_Toc511034025)

[Vaardighede wat vereis word i](#_Toc511034026)

[Wat moet die leerder vooraf geleer word? ii](#_Toc511034027)

[Wanpraktyk ii](#_Toc511034028)

[Leerder se verklaring van egtheid van die PAT ii](#_Toc511034029)

[Rol van die onderwyser iii](#_Toc511034030)

[Toesig/Gekontroleerde toestande iii](#_Toc511034031)

[Bestuur van die PAT iv](#_Toc511034032)

[Bewyse van assessering iv](#_Toc511034033)

[Vereistes iv](#_Toc511034034)

[Versuim v](#_Toc511034035)

# Wat is die PAT?

Die PAT is ’n programmatuur-ontwikkelingsprojek wat jou die geleentheid gee om bewys te lewer van jou programmeringsvaardigheid asook jou begrip van die onderlinge verband tussen die verskillende areas van oplossingsontwikkeling.

Daar word ook van jou verwag om bewys te lewer van jou kennis en begrip van die programmatuur-ontwikkelingsiklus, deur middel van ontleding, ontwerp, kodering en toetsing. Jy moet ook bewys dat jy die programmatuur-ontwerpgereedskap en tegnieke wat jy geleer het, effektief kan gebruik.

Jy moet die volgende inhandig:

* ’n Verslag (Fase 1) waarin jy
  + ’n kort beskrywing gee van die probleem wat jy moet oplos
  + ‘n bespreking gee van die navorsing/ondersoek wat jy gedoen het oor die projek
  + ’n kort beskrywing gee van die doel en omvang van die projek
* ’n dokument met ‘n uiteensetting van die programontwerp (Fase 2)
* ’n Werkende Delphi-program wat die beplande oplossing implementeer (Fase 3)

**Nota:**

Daar word van jou verwag om jou finale program tydens ’n ondervraging sessie te demonstreer en te bespreek.

## **Puntetoekenning**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fase** | **Ontwikkelingsfase** | **Maks. Punt** | **%** |
| **Fase 1** | Ontleding | **24** |  |
| **Fase 2** | Ontwerp | **44** |  |
| **Fase 3** | Kodering, Toetsing en Kompleksiteit | **60** |  |
| **Algemeen** | Finale produk en indruk | **22** |  |
| **Totaal** | | **150** | **100** |

Die PAT tel 25% van jou finale punt vir IT, daarom is dit van uiterste belang dat jy sal poog om werk van hoë gehalte te lewer.

Die PAT is ’n verpligte komponent van die finale eindeksamen in IT.

Jy moet jou PAT voltooi voordat jy met die Graad 11 eindeksamen begin. Indien jy nie die PAT, of enige gedeelte daarvan, indien nie, sal ’n nul (“0”) vir die PAT-komponent toegeken word, of vir die gedeeltes van die PAT wat nie ingedien is nie.

# Onderwerp

***Stokperdjies***

'n Stokperdjie is 'n aktiwiteit wat gereeld beoefen word vir genot, gewoonlik gedurende vrye tyd. (Wikipedia).

Baie mense met stokperdjies wil graag ‘n toepassing hê om te help met sommige aspekte van hul stokperdjie, bv. organisering / dokumentering.

Stokperdjies kan as voorbeeld die volgende insluit:

* Items versamel (bv. Versamel / luister na musiek)
* Fotografie (organiseer, dokumenteer, vind foto's)
* Draf (roetes, marathons, baanfiksheid, ens.)
* Video speel (spelers, tellings, digitale bates, ens.)
* Ens.

Idees vir jou program:

* 'n Databasis om musiek en video of films by te voeg, te organiseer, te speel en te vind (bv. Soos iTunes / Netflix).
* 'n Databasis om foto's by te voeg, te organiseer en te vind met behulp van datums, ligging, ens.
* Ens.

U finale program moet bestaan ​​uit een enkele, logies verwante stuk sagteware (nie aparte programme nie). Projekte wat uit twee of meer onverwante programme bestaan, sal slegs punte ontvang vir een van die dele, aangesien slegs een van die programme as die werklike projek beskou sal word.

Die kwaliteit van die programmeringskode wat data suksesvol manipuleer, volgens die gebruiker se behoeftes, sal jou projekpunt aansienlik beïnvloed. Kwantiteit sal nie die verskeidenheid, doeltreffendheid en kwaliteit vervang nie.

# Oorsig

## **Fase 1 – Analise**

Die doel van Fase 1 is om te bepaal ***wat*** gedoen moet word en ***wat*** is die gebruiker se vereistes:

* Identifiseer die probleem
* Die taak in jou eie woorde te definieer
* Ondersoek/vors die onderwerp/scenario/die (potensiële) gebruikers na om feite betreffende die aard/funksionaliteit van die program wat jy wil ontwikkel, in te win.
* **Definieer die taak deur die volgende te bepaal:**
  + ***Wat*** die stelsel sal doen
  + ***Hoe*** die stelsel gebruik sal word
  + ***Wie*** die stelsel sal gebruik

## **Fase 2– Ontwerp**

Die doel van Fase 2 is om te bepaal ***hoe*** die program/stelsel aan die vereistes sal voldoen en om ’n oplossing te beplan en te ontwerp vir die probleem.

* Gebruik die beskikbare programontwerpgereedskap om die vereistes duidelik te stel en die oplossing te ontwerp. Dui duidelik die logiese programvloei en navigasie tussen skerms aan.
* Ontwerp die databasis en ander datastrukture wat gebruik gaan word.
* Ontwerp die GGK(e) en ander datavaslegging meganismes.
* Identifiseer watter verwerking nodig is en definieer hoe die verwerking gedoen word.
* Dui aan wat die afvoer is en ontwerp afvoerkomponente/meganismes.

## **Fase 3 – Kodering en Toetsing**

Die doel van Fase 3 is die implementering van die ontwerp deur die kode te skryf en die program te toets:

* Skryf die programmeringskode om die ontwerp te implementeer en voltooi die program.
* Toets en ontfout die program.
* Voeg kommentaar by om stukke kode te verduidelik.
* Skryf projeknotas vir die program.
* Demonstreer jou program en beantwoord vrae omtrent die program en die kode tydens ’n ondervragingsessie.

# PAT-Vereistes

Die projek moet die volgende, toepaslik geïntegreer, insluit:

* Databasismanipulasie deur van hoëvlak programmeertaalkonstruksie gebruik te maak
* ’n Tekslêer vir toevoer-afvoer-doeleindes, bv. om na datastrukture te lees, verslae op te stel
* ’n Multi-vorm/multi-skerm GGK (*GUI*) met goeie funksionaliteit en bruikbaarheid, gebaseer op goeie MRI-beginsels (HCL)
* Manipulasie/transformasie van data deur
  + Wiskundige/statistiese prosesse
  + String/teks-prosesse

**Databasis**

Die databasiskomponent

* moet oopgemaak en gemanipuleer word deur jou eie kode
* mag nie slegs die stoor/onttrekking van data en moontlike sinnelose verwerking deur die pakket self, met byna geen eie kode (wat lei tot ’n eenvoudige oplossing), behels nie
* moet insluit voldoende hoeveelheid data en ’n verskeidenheid veld-tipes gebruik

**Tekslêer**

Jou program moet toevoer aanvaar vanaf ’n tekslêer, bv. om data wat elders gestoor is te hanteer. Dit moet meganismes insluit om data tussen die databasis en die tekslêer(s) oor te dra en te verwerk.

Die data van die tekslêer kan gebruik word om

* berekeninge en manipulasies in kombinasie met data in die databasis te doen
* bestaande rekords by te werk (voeg rekords by, vee rekords uit, werk rekords by, verander rekords)

Ten minste ***een*** verslag moet in tekslêer-formaat verskaf word.

**GGK (*GUI*)**

Die grafiese gebruikerskoppelvlak (GGK) moet funksioneel wees en gebaseer wees op goeie MRI-beginsels.

Die GGK moet ten minste

* drie vorms/skerms hê
* een komponent dinamies skep

**Veranderlikes en datastrukture**

* Gebruik toepaslike veranderlikes en datastrukture
* Oorweeg die reikwydte van die veranderlikes deeglik (lokaal vs. globaal)

**Modulêre Programmering**

Inkorporeer jou eie metodes/prosedures/funksies, bv. om data te valideer of om data te verander/manipuleer.

**Verdere vereistes:**

*Pas goeie programmeringsbeginsels en tegnieke toe.*

* Beskrywende name vir veranderlikes, datastrukture, velde, komponente, ens.
* Goed-gestruktureerde, leesbare kode
* Gebruik kommentaar om dele van die kode te verduidelik, veral oor die manier waarop veranderlikes/datastrukture en afvoer-komponente/veranderlikes gebruik word.

*Skryf projeknotas*

* Verduidelik hoe die program gebruik moet word
* Beskryf enige foute of probleme waarvan jy bewus is

Projeknotas kan geskryf word as ’n hulpfunksie wat deel vorm van die program

**Algemene programmeringsaspekte wat assesseer word:**

* Programmeringstyl
* Grafiese gebruikerskoppelvlak (GGK/*GUI*)
* Gebruik van mens-rekenaar-interaksie (MRI) en programmatuurontwikkelingsbeginsels.
* Funksionaliteit van die program
* Kundigheidsvlak van programmering
* Robuustheid van die program, insluitend die gebruik van defensiewe programmeringstegnieke
* Of die projek aan die oorspronklike doelwitte voldoen
* Interne dokumentasie om dele van die program te verduidelik

# woordelys

Sien **Bylaag C** vir ‘n lys van Beskrywing van terme.

Sien **Bylaag D** vir ‘n voorbeelde.

# Wat jy BEnodig om die PAT te VOLTOOI

Om die taak te voltooi, sal jy die volgende nodig hê:

* Delphi programmeringsprogrammatuur, insluitend ’n GUI geïntegreerde ontwikkelings-omgewing (IDE)
* ’n Kantoorpakket met die volgende programmatuur
  + Woordverwerkingsprogrammatuur
  + Databasisprogrammatuur
* Internettoegang om data en inligting te vind
* Toegang tot ander bronne soos gedrukte media (bv. tydskrifte, koerante, brosjures, handboeke) of ander elektroniese materiaal soos e-boeke, e-artikels
* Toegang tot fasiliteite waarmee jy hardekopieë na elektroniese dokumente kan omskakel, bv. skandeerder, digitale kamera
* Bergingsmedia om jou werk elektronies te stoor en te rugsteun, bv. flitsskyf, herskryfbare CD/DVD , OneDrive, ens.

# Wangedrag

Omdat die PAT ’n individuele projek is en deel is van jou finale promosiepunt, mag jy nie:

* Hulp van onder kry sonder om die nodige erkenning te gee nie
* Werk indien wat nie jou eie is nie, bv. programmeringskode wat deur ’n ander persoon ontwikkel is
* Jou PAT-werk aan ander leerders uitleen nie
* Ander leerders toelaat om jou werk te bekom of te gebruik nie (dit beteken nie dat jy nie boeke mag uitleen of leen van ’n ander leerder nie, maar jy mag nie plagiaat pleeg deur ander leerders se navorsing of kode te gebruik nie)
* Werk insluit wat direk uit boeke, vanaf die internet of ander bronne gekopieer is nie, sonder om nodige erkenning daaraan te verleen nie (dit mag ook nie 20% van die werk wat jy indien oorskry nie)

Die bostaande optredes kom neer op wangedrag waarvoor jy gepenaliseer sal word.

# Versuim

Jy sal die geleentheid gegee word om enige uitstaande werk in te dien of om jouself aan te meld om die PAT te doen, soos uiteengesit.

Indien jy versuim om die PAT-vereistes na te kom, sal ’n punt van nul (“0”) toegeken word vir enige uitstaande dele van die PAT of vir die PAT in geheel (indien alles uitstaande is).

# Instruksies vir Fase 1

Die doel van Fase 1 is om

* die taak in jou eie woorde te definieer
* navorsing te doen om duidelike begrip van die probleem/taak ophande te ontwikkel
* jou navorsing te ontleed om vas te stel wat die stelsel moet doen, hoe die stelsel gebruik gaan word en wie die gebruikers gaan wees – indien moontlik raadpleeg die gebruikers om die vereistes/funksionaliteit van die program vas te stel

### Definieer die Taak

Die doel is om ’n oorsigtelike beeld van die doel en omvang van die projek te gee, maar ***nie detail nie***.

In jou eie woorde, gee ‘n kort, algemene beskrywing (±150 woorde) van die probleem/taak en hoe die projek dit sal oplos. Met ander woorde, die beskrywing moet die leerder of onderwyser oortuig dat

* jy die behoeftes/tekortkominge/probleme verstaan
* jou oplossing die behoeftes/tekortkominge/probleme sal aanspreek

Hulle moet ook daarna uitsien om jou program te gebruik. Jou beskrywing moet ook die PAT vereistes/spesifikasies aanspreek.

### Doen Navorsing

**Die doel is om agtergrondinligting en feite in te samel oor die geïdentifiseerde taak en die aard van die program wat jy ontwikkel.**

Jou navorsing moet jou help om:

* die onderwerp/scenario te verstaan
* duidelikheid te kry betreffende die tipe/aard van die program wat jy moet ontwikkel
* waar moontlik, na bestaande oplossings te kyk en idees te versamel
* te verstaan watter spesifieke tipe program geskik is vir die projek

**Die uitkoms van die navorsing is ’n verslag**(±600 woorde/2 bladsye) wat byvoorbeeld aandui wat ’n program vir die probleem moontlik kan insluit, ooreenkomste, ontbrekende fasiliteite/kenmerke en algemene werking/vloei van bestaande oplossings**. Die verslag moet in duidelike, ondubbelsinnige taal geskryf word.**

### Doen die Ontleding

Die doel is om

* te bepaal ***wat*** die stelsel gaan doen
* te bepaal ***hoe*** die stelsel gebruik gaan word
* te bepaal ***wie*** die stelsel gaan gebruik

### Handig In

**Wanneer jy Fase 1 van die projek voltooi het, handig ’n gedetailleerde verslag in wat die volgende dek:**

* Probleemidentifisering
* Navorsing/ondersoek van die onderwerp/scenario, die potensiële gebruikers en feite betreffende die aard/funksionaliteit van die program wat jy beplan om te ontwikkel

Dit moet bewyse van navorsing insluit met duidelike verwysings

* ’n Gedetailleerde taakdefinisie wat die volgende aandui:
  + WAT die stelsel sal doen
  + HOE die stelsel gebruik sal word
  + WIE die stelsel sal gebruik

Sluit in jou egtheidsverklaring vir Fase 1 (**Bylaag B**)

Sien **Bylaag A** vir die struktuur van die verslag.

**NOTA:** Jou verslag moet ***NIE*** insluit dat jy ’n Delphi-program moet insluit ***nie***, dat dit gebruikersvriendelik moet wees nie of dat dit ’n databasis moet insluit nie. Hierdie is voor die handliggende vereistes. Fokus op die ***ENKELE***, spesifieke probleem wat jy gekies het.

# Instruksies vir Fase 2

Die doel is om

* te bepaal ***HOE*** jy te werk sal gaan om die probleem op te los en om die detail te beplan
* ’n plan te bied wat ’n hoëvlak oorsig gee betreffende die konstruksie van die oplossing met behulp van verklarende pseudokode/diagramme (of geskikte alternatiewe). Die plan moet die hoof blokke aantoon binne die voorgestelde oplossing.
* Spesifiseer en te dokumenteer ‘n algemene ontwerp, wat die vereistes nakom, deur gebruik te maak van programontwerp-gereedskap soos TVA-diagramme, TOG-diagramme (*TOE charts*) en vloeidiagramme met beskrywende notas te gebruik.

### Basiese Oplossingsontwerp

Gebruik ten minste twee verskillende programontwerpgereedskap soos TVA-diagramme, TOG-diagramme, vloeidiagramme om die programvloei en navigasie tussen skerms duidelik aan te dui/te ontwerp. Dit moet die verskillende vorms en/of objekte wat gebruik gaan word, duidelik identifiseer.

### Data-woordeboek

Ontwerp ’n databasis wat as databron sal dien en wat data kan voorsien wat deur wiskundige/ statistiese prosesse, met behulp van programkode, gemanipuleer/verander kan word.

Dui die velde, veldtipes en primêre sleutel duidelik aan.

Beskryf die rol van die databasis in die program.

Die Delphi-program moet die inhoud van die databasistabel manipuleer, bv. rekords bywerk/ verander/skrap/byvoeg, resultate van navrae voorsien, verslae as ’n produk van verwerking/manipulering van data, voorsien, ens.

Beplan en ontwerp ander datastrukture wat gebruik gaan word. Dit sluit spesifieke veranderlikes, skikkings en tekslêers in. Dui duidelik aan hoe en waar hierdie datastrukture gebruik gaan word.

### Datavaslegging (Toevoer)

Ontwerp ’n GGK met inagneming van goeie rekenaar-mens-koppelvlak (*MRI*)-beginsels wat ook foute verhoed wat voorkom as gevolg van ongeldige toevoer en wat die hoeveelheid inligting wat ’n gebruiker moet voorsien, tot die minimum beperk.

Gebruik TOG-diagram(me) en dui die verantwoordelikhede en gebeurtenisse vir alle kontroles aan.

Gebruik *MRI*-ontwerpbeginsels en ontwerp ’n GGK wat die volgende in ag neem:

* Die gebruiker – tipe en konteks
* Gebruiker se vereistes/behoeftes, bruikbaarheid (*usability*)
* Dialoog – moet toepaslik, eenvoudig en duidelik wees
* Gebruik en aanbieding van ikone – gepaste keuses, goed geplaas met duidelike doel
* Kleur – gebruik van kleur en kleurkombinasies
* Terugvoer – netjies, duidelik en goed aangebied
* Hulpvaardige foutboodskappe
* Uitgange – duidelik gemerk, korrek geplaas
* Kortpadsleutels
* Vloei van inligting op die skerm – bo na onder en links na regs
* Sinvolle gebruik van die spasie op die skerm

Voorsien voorbeelde van beplande datavaslegging en datatoevoer-ontwerp (prototipe skermkopieë mag gebruik word maar moet beskrywende notas bevat) en van beplande, geldige afvoer-ontwerp.

Beplan en ontwerp hoe datavalidering en foutvang vir alle toepaslike komponente hanteer sal word.

### Dataverwerking

Gebruik die ontwerpgereedskap en identifiseer en lys die verwerkings wat gedoen moet word. Ontwerp die nodige algoritmes om die verlangde verwerkings te doen.

Voorsien:

* ’n Gedetailleerde lys van alle verwerking wat gedoen moet word
* Algoritmes/metodes wat gebruik gaan word om die verlangde verwerking te doen. Dit kan die volgende insluit:
  + Spesifieke formules en/of funksies wat gebruik gaan word
  + Pseudokode-oplossings en diagramme

### Afvoer

Gebruik die ontwerpgereedskap en identifiseer en lys die afvoer wat benodig gaan word. Selekteer/ontwerp die nodige afvoerkomponente en hoe die afvoer geformateer moet word.

Voorsien:

* ’n Gedetailleerde lys van alle nodige afvoer
* Afvoerkomponente wat gebruik gaan word en hoe die afvoer geformateer gaan word

### Handig In

Sodra jy Fase 2 van die projek voltooi het, handig die volgende in:

’n Dokument met die volgende

* Basiese ontwerp – gebruik ten minste twee programontwerp gereedskapitems
* Datawoordeboek – insluitend gedetailleerde databasisontwerp
* Datavaslegging –GKK-ontwerp, TOG-diagram, validerings- en foutvangtegnieke
* Dataverwerking – watter verwerking gedoen moet word en hoe dit verwerk moet word
* Data-afvoer – afvoer benodig, komponente betrokke en formaat van afvoer

Sluit ook jou egtheidsverklaring vir Fase 2 in (**Bylaag B**)

# Instruksies vir Fase 3

Die doel is om

* jou ontwerp te implementeer deur van geskikte programmatuur (programmeertaal, databasis-programmatuur, GOO (*IDE*), ens.) en tegnieke gebruik te maak om ’n oplossing vir die probleem te bied.
* die program te demonstreer en vrae te beantwoord oor die program, die proses en die kode

### Ontwikkel die Databasis

Implementeer die ontwerp en skep die databasis deur toepaslike tegnieke te gebruik.

Maak seker dat die databasis korrek koppel met die program en integreer met die program sodat daar betekenisvolle en effektiewe wisselwerking tussen die program en die databasis plaasvind, op so ’n wyse dat dit die oplossing ondersteun.

### Ontwikkel die GGK (*GUI*)

Implementeer die ontwerp deur die GGK(s) te ontwikkel. Gebruik toepaslike komponente wat die gebruiker se gebruik en navigasie sal vergemaklik. Die gebruiker moet ’n aangename ervaring hê wanneer hy/sy die program gebruik.

### Skryf die Kode

Gebruik die beplanningsdokumente van Fase 1 en Fase 2 en skryf die kode vir al die eenhede/dele.

Gebruik goeie programmeringstegnieke en -strukture.

Implementeer effektiewe algoritmes en goeie defensiewe programmeringstegnieke om ’n robuuste program te verseker.

Dokumenteer die kode sodat ander mense in staat sal wees om die program te interpreteer en sal verstaan wat individuele stukke kode doen.

### Toets die Program/Stelsel

Voer toetse uit om die volgende te bepaal:

* Die funksionaliteit van die program/stelsel – om te bevestig dat die program/stelsel aan die vereistes voldoen (gebruikers aanvaardingstoetse)
* Of eenhede met kode (enkel funksies, prosedures, koppelvlak(ke), ens. – een kenmerk op ’n slag) korrek werk (eenheidstoetsing)

Toets die program/stelsel deur duidelik gedefinieerde, tipiese data, foutiewe data en grensdata (ekstreme/uiterstes) te gebruik.

Vergelyk die toetsresultate met die verwagte resultate om sukses of mislukking te bepaal.

Ontfout waar nodig.

### Dokumenteer die Program

Gebruik enige geskikte fasiliteit van die programmeringstaal en skryf projeknotas wat vir die gebruiker toeganklik is. Dit moet verduidelik hoe om die program te gebruik.

Die notas moet ook enige programfoute of probleme wat aan jou bekend is, beskryf.

Voeg kommentaar in om stukke kode te verduidelik.

### Handig In

Wanneer jy Fase 3 voltooi het, handig die volgende in:

* Die voltooide Delphi-projek, insluitend die kommentaar en projeknotas.
* Die egtheidsverklaring vir Fase 3 (**Bylaag B**)

### Ondervraging

Demonstreer die program vir evaluering en ondervraging.

Riglyne vir die demonstrasie van die program:

* Die onderwyser sal datums en tye vir die demonstrasies skeduleer. Daar sal ongeveer 15 minute per projek toegelaat word.
* Jy moet voor die demonstrasie alle dokumentasie inhandig – ten minste een week voor die tyd.
* Die demonstrasie word elektronies op ’n rekenaar gedoen.
* Vir evalueringsdoeleindes moet jy jou program uitvoer en al die kenmerke van die program aan jou onderwyser uitwys.
* Die onderwyser kan van jou vereis om toetsprosedures uit te voer om seker te maak dat die hele program korrek werk.
* Die onderwyser kan gedurende die demonstrasie die merkblad vir Fase 3 as ’n riglyn gebruik en punte dienooreenkomstig toeken.
* As deel van die demonstrasie, sal die onderwyser op ’n ewekansige manier dele van die programkode identifiseer en jou vra om die doel en werking daarvan te verduidelik. Dit word gedoen om seker te maak dat jy die kode self geskryf het. ’n Soortgelyke prosedure sal tydens moderering gevolg word. Indien jy nie die kode in die projek kan verduidelik nie, kan geen punte vir die projek toegeken word nie.
* Jy moet die elektroniese kopie van die projek wat jy gedemonstreer het inhandig. Die onderwyser sal hierdie kopie gebruik om enige uitstaande punte toe te ken om sodoende die punt te finaliseer.

### Sterkte!

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fase 1: Analise Naam van leerder:** | | | | | | | | |
|  | | | | | | | | |
| **Ondersoek** | **4** | **3** | **2** | **1** | **0** |  |  | |
| **Probleem-identifikasie** | Duidelike beskrywing van die doel van die program | Aanvaarbare beskrywing van die doel van die program | Beperkte beskrywing van die doel van die program | Swak beskrywing van die doel van die program | Geen duidelike doel gedefinieer nie | **4** |  | |
| **Navorsing** | Uitgebreide navorsing gedoen. Navorsing is relevant Bewys van eie navorsing en/of Ten minste 3 verwysings van ander bronne (web, boeke, ens.) Ten minste 2 bestaande stelsels nagevors | Uitgebreide navorsing gedoen. Beperkte relevansie Ten minste 2 verwysings van ander bronne (web, boeke, ens.) Ten minste 1 bestaande stelsel nagevors | Groot hoeveelheid navorsing gedoen, maar nie altyd relevant. Beperkte verwysing/bewyse | Beperkte navorsing en/of verwysing. | Geen navorsing/verwysings ingesluit nie | **4** |  | |
| **Taak/probleemanalise en definisie** | **4** | **3** | **2** | **1** | **0** |  |  | |
| **Wat sal die stelsel doen** | Duidelike beskrywing wat alle detail dek | Duidelike beskrywing met mindere detail wat ontbreek | Volledige beskrywing, maar ontbreek duidelikheid | Onvolledige beskrywing met beperkte gebruik | Verkeerd, irrelevant of nie beskryf nie | **4** |  | |
| **Hoe sal die stelsel gebruik word** | Duidelike beskrywing wat alle detail dek | Duidelike beskrywing met mindere detail wat ontbreek | Volledige beskrywing, maar ontbreek duidelikheid | Onvolledige beskrywing met beperkte gebruik | Verkeerd, irrelevant of nie beskryf nie | **4** |  | |
| **Wie gaan die stelsel gebruik** | Duidelike beskrywing wat alle detail dek | Duidelike beskrywing met mindere detail wat ontbreek | Volledige beskrywing, maar ontbreek duidelikheid | Onvolledige beskrywing met beperkte gebruik | Verkeerd, irrelevant of nie beskryf nie | **4** |  | |
| **Algehele kwaliteit** | **4** | **3** | **2** | **1** | **0** |  |  | |
| **Waarde van analise as deel van projekontwerp** | Uitstekende, duidelike riglyne vir die projek gebaseer op ondersoek.  Probleem duidelik gedefinieer - toon duidelike begrip en insig | Genoegsame riglyne – nie altyd op ondersoek gebaseer.  Probleem nie duidelik met betrekking tot sommige aspekte nie  Toon voldoende insig | Beperkte riglyne – nie altyd op ondersoek gebaseer.  Basiese ontleding van die probleem  Beperkte insig | Vae riglyne met min relevansie tot die ondersoek.  Probleem nie duidelik ontleed nie  Minimale insig | Nie gebaseer op ondersoek of onderwerp irrelevant.  Geen waarde met begrip van probleem nie | **4** |  | |
| **Totaal** | | | | | | **24** | |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Fase 2: Ontwerp Naam van leerder:** | | | | | | | |
| **BASIESE ONTWERP** |  |  |  |  |  |  |  |
| **Programontwerp-gereedskap – enige TWEE – dui asseblief aan watter evalueer is** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Gereedskapsitem 1**   * TVA-tabel * Gebruikersgeval-diagram * Klasdiagram * Entiteitsverwantskaps-diagram (ERD) | Uitstekende gebruik van gereedskap wat die program duidelik definieer | Gereedskap goed genoeg gebruik om die program te definieer | Gereedskap gebruik met tekortkominge. Program-definisie nie duidelik nie | Gereedskap nie duidelik gekonstrueer nie | Geen gereedskap, verkeerde gebruik van gereedskap, irrelevant tot toepassing | **4** |  |
| **Gereedskapsitem 2**   * TVA-tabel * Gebruikersgeval-diagram * Klasdiagram * Entiteitsverwantskaps-diagram (ERD) | Uitstekende gebruik van gereedskap wat die program duidelik definieer | Gereedskap goed genoeg gebruik om die program te definieer | Gereedskap gebruik met tekortkominge. Program-definisie nie duidelik nie | Gereedskap nie duidelik gekonstrueer nie | Geen gereedskap, verkeerde gebruik van gereedskap, irrelevant tot toepassing | **4** |  |
| **DATAWOORDEBOEK** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Keuse van datastrukture:**  Skikkings, tekslêers, veranderlikes, ens.  **(*Hoe*** data gestoor word**)** | Verskeidenheid datastrukture wat almal bydrae tot die oplossing en duidelik gedefinieer | Verskeidenheid datastrukture maar is nie altyd die beste oplossing vir die probleem nie of nie duidelik gedefinieer nie | Datastrukture sluit nie slegs toepaslike veranderlikes in nie, maar ook ander strukture soos tekslêers en skikkings | Definisie dek basiese veranderlikes met toepaslike tipes wat nodig is vir die oplossing | Geen of swak definisie van datastrukture wat gebruik word | **4** |  |
| **Databasisontwerp**   * Alle velde dra by tot die oplossing * Tipe en grootte van velde goed gekies | Voldoen aan alle databasis-ontwerpvereistes | Goeie databasisontwerp met minimale tekortkominge | Gemiddelde ontwerp met verskeie tekortkominge | Databasisontwerp gedoen, maar met beperkte waarde | Geen databasis of verkeerd of irrelevant | **4** |  |
| ***DATAVASLEGGING*** | **4** | **3** | **2** | **1** | **0** |  |  |
| **GUI-ontwerp**   * Ontwerp geskik vir program se gebruik * Toepaslike komponente * Gemak van gebruik, logiese vloei * Duidelik gemerkte navigasie * Vriendelike dialoog/Hulp | Goeie GGK-ontwerp, bevat byna al (ten minste 5) van die beginsels vir alle koppelvlakke | Bevredigende GGK-ontwerp, bevat meeste (ten minste 4) van die beginsels vir alle koppelvlakke | Beperkte *GUI* ontwerp, bevat slegs 50% (ten minste 3) van die beginsels vir alle koppelvlakke | Swak *GUI* ontwerp, bevat slegs minder as 50% (minder as 3) van die beginsels vir alle koppelvlakke | GKK is nie funksioneel nie of ondersteun nie die voorgestelde gebruik nie | **4** |  |
| **TOG-diagram** | Duidelike beskrywing van die verantwoordelikheid en gebeurtenisse vir alle komponente | Verantwoordelikheid en gebeurtenisse vir meeste komponente duidelik beskryf | Verantwoordelikheid en gebeurtenisse vir meeste komponente nie duidelik beskryf nie | Verantwoordelikheid en gebeurtenisse swak beskryf | Geen TOG-diagram of nie funksioneel nie | **4** |  |
| **Validering/Foutvang**   * Validering en/of foutvang vir relevante toevoer * Geassosieerde foutboodskappe | Duidelike beskrywing van toepaslike en effektiewe tegnieke om te verseker dat slegs geldige data ingevoer word | Tegnieke om te verseker dat slegs geldige data ingevoer word, is meestal toepaslik en effektief | Validering en/of foutvang beperk en soms ontoepaslik/nie betekenisvol nie | Validering en/of foutvang swak beskryf of ontoepaslik/nie betekenisvol nie | Geen poging tot validering | **4** |  |
| **DATAVERWERKING** | **4** | **3** | **2** | **1** | **0** |  |  |
| ***WATTER*** verwerking gedoen gaan word | Lys duidelik alle verwerking/manipulasie wat benodig word | Een of twee verwerking/manipulasie nie korrek gelys nie | Meeste verwerking/manipulasie korrek gelys. Beperkte detail | Baie verwerking/manipulasie nie duidelik gelys nie | Meeste verwerking/manipulasie nie gelys nie |  |  |
| ***HOE*** verwerking bestuur gaan word (algoritmes, formules, ens.) | Duidelike beskrywing hoe alle verwerking/manipulasie gedoen moet word | Een of twee verwerking/manipulasie nie duidelik beskryf nie | Some verwerking/ manipulasie nie duidelik beskryf nie. Meestal korrek | Baie verwerking/ manipulasie nie duidelik beskryf nie. Baie foute | Verwerking/ manipulasie nie beskryf nie, verkeerd of irrelevant |  |  |
| **AFVOER** | **4** | **3** | **2** | **1** | **0** |  |  |
| ***Watter*** afvoer voorsien sal word | Duidelike beskrywing van alle afvoer kragtens die data wat voorsien moet word en hoe afvoer-bronne gebruik sal word | Geringe tekortkominge in beskrywings. Een of twee afvoere nie duidelik beskryf nie. | Beperkte beskrywing. Meer as twee afvoere nie duidelik beskryf nie. | Swak beskrywing. Meeste afvoere nie duidelik beskryf nie. | Afvoervereistes nie beskryf nie, verkeerd of irrelevant |  |  |
| **Formaat van afvoer**  **(*Hoe*** afvoer aangebied sal word**)** | Duidelike beskrywing van alle afvoer kragtens hoe dit sal vertoon, formaat, tipe en komponente vir afvoer | Geringe tekortkominge in beskrywings. Een of twee afvoere nie duidelik beskryf nie. | Beperkte beskrywing. Meer as twee afvoere nie duidelik beskryf nie. | Swak beskrywing. Meeste afvoere nie duidelik beskryf nie. | Afvoervereistes nie beskryf nie, verkeerd of irrelevant |  |  |
| **Totaal** | | | | | | **44** |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fase 3: Kodering Naam van leerder:** | | | | | | | | |
| **DATASTRUKTURE** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Datastrukture**  (Gebruikers-gedefinieerd, uitgesluit dB) | Ten minste twee datastrukture (bv. skikking en tekslêer) korrek en toepaslik gebruik | Ten minste twee datastrukture (bv. skikking en tekslêer) met geringe tekortkominge gebruik | Ten minste een datastruktuur (bv. skikking of tekslêer) korrek en toepaslik gebruik | Ten minste een datastruktuur (bv. skikking of tekslêer) gebruik maar nie altyd korrek of toepaslik nie | Heeltemal ontoepaslik of verkeerd of nie gebruik nie | **~~4~~** |  |
| **Implementering van databasis-ontwerp** | Databasisontwerp korrek geïmplementeer met ten minste twee tabelle, geskikte velde, datatipes en –groottes.  Groot/voldoende volume data gebruik. | Databasisontwerp korrek geïmplementeer met ten minste twee tabelle, geskikte velde, datatipes en –groottes.  Beperkte volume data gebruik. | Ten minste twee tabelle gebruik, maar databasis-ontwerp nie behoorlik geïmplementeer nie. Foute in velde, datatipes en –groottes nie alles geskik nie OF  Klein hoeveelhede data gebruik | Gebruik van toepaslike databasis  Een tabel  Min toepaslike velde, data-tipes en -groottes | Heeltemal ontoepaslik of verkeerd of nie gebruik nie | **4** |  |
| **TOEVOER** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Toevoer**  (databasis, tekslêer, gebruikers-toevoer) | Al drie tipes gebruik.  Toepaslikste, effektiefste toevoerstrategieë (dB, tekslêer, gebruiker) in alle gevalle gebruik | Al drie tipes gebruik, maar in een geval kon ’n toepasliker of effektiewer strategie gebruik word | In twee gevalle kon ’n toepasliker of effektiewer strategie gebruik word | In meer as twee gevalle kon ’n toepasliker of effektiewer strategie gebruik word | Heeltemal ontoepaslik of oneffektief | **4** |  |
| **GKK-ontwerp**  Toevoer-komponente / HCI beginsels | Toepaslikste keuse vir alle komponente.  Uitstekende gebruik van skermwenke, terugvoer in alle gevalle  Uitstekende gebruik van effekte/kleur/kortpaaie/ikone/gereedskapswenke, ens. | Toepaslikste keuse vir kom-ponente in meeste gevalle.  Goeie gebruik van skerm-wenke, terugvoer  Goeie gebruik van effekte/ kleur/kortpaaie/ ikone/ gereedskapswenke, ens. | Verskeie komponente is nie die toepaslikste keuse  Bevredigende gebruik van skermwenke, terugvoer  Bevredigende gebruik van effekte/kleur/kortpaaie/ ikone/gereedskapswenke, ens. | Min toepaslike komponente.  Beperkte of geen gebruik van skermwenke, terugvoer.  Beperkte of geen gebruik van effekte/kleur/kortpaaie/ ikone/gereedskapswenke, ens. | Swak GKK ontwerp, min/geen aandag aan HCI-beginsels nie | **4** |  |
| **Validering/ Foutvang** | Uitgebreide validering/fout-vang vir toepaslike toevoer  Duidelike, toepaslike fout-boodskappe en uit-sonderingshanterings-meganismes | Verskeidenheid validering/ foutvang vir toepaslike toevoer  Meestal duidelike, toepaslike foutboodskappe en uit-sonderingshanterings-meganismes | Beperkte validering/ foutvang  Foutboodskappe en uit-sonderingshanterings-meganismes soms ontoe-paslik/nie betekenisvol nie | Validering/foutvang swak beskryf of ontoepaslik/nie betekenisvol nie | Geen poging tot validering | **4** |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Verwerkings algoritmes** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Korrektheid van algoritmes** | Alle oplossingsalgoritmes is toepaslik, werk korrek en voldoen aan al die verwerkingsvereistes | Toepaslike oplossings-algoritmes wat korrek werk maar een verwerkings-vereiste nie vervul nie | Meeste oplossingsalgoritmes is toepaslik, werk korrek en voldoen aan meeste van die verwerkingsvereistes | Oplossingsalgoritmes is meestal onvoldoende of werk meestal nie korrek nie of verwerkingsvereistes nie vervul nie. | Heeltemal onvoldoende of werk nie korrek nie | **4** |  |
| **Effektiwiteit van algoritmes** | Alle algoritmes is die effektiefste oplossing.  Algoritmes gebruik goeie programmeringstegnieke  Effektiewe, modulêre ontwerp met korrekte gebruik van eie funksies en prosedures. | Meeste algoritmes is die effektiefste oplossing.  Algoritmes gebruik aanvaar-bare programmeringstegnieke  Beperkte modulêre ontwerp met korrekte gebruik van eie funksies en prosedures. | Beperkte effektiwiteit van algoritmes.  Min algoritmes gebruik goeie programmeringstegnieke  Swak modulêre ontwerp met beperkte gebruik van eie funksies en prosedures. | Swak effektiwiteit van algoritmes.  Algoritmes gebruik nie goeie programmeringstegnieke nie  Poging aangewend om eie funksies en prosedures te gebruik. | Heeltemal onvoldoende of nie gebruik nie | **4** |  |
| **Gebruik van kreatiewe/ komplekse kode** | Uitgebreide gebruik van teken/animasie/grafieke/ kaarte/tyd/komplekse bere-keninge/komplekse string-manipulasie | Voldoende gebruik van teken/animasie/grafieke/ kaarte/tyd/komplekse bere-keninge/komplekse string-manipulasie | Beperkte gebruik van teken/animasie/grafieke/ kaarte/tyd/komplekse bere-keninge/komplekse string-manipulasie | Poging tot gebruik van teken/animasie/grafieke/ kaarte/tyd/komplekse bere-keninge/komplekse string-manipulasie | Heeltemal onvoldoende of nie gebruik nie | **4** |  |
| **Databasis – Hoëvlak taal interaksie** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Effektiewe gebruik van data-navrae** | Uitstekende gebruik van kode-konstrukte vir beskou en navigasie van data  Goeie gebruik van kode om te filter/sorteer/soek | Goeie gebruik van kode-konstrukte vir beskou en navigasie van data  Voldoende gebruik van kode om te filter/sorteer/soek | Voldoende gebruik van kode-konstrukte vir beskou en navigasie van data  Beperkte gebruik van kode om te filter/sorteer/soek | Beperkte gebruik van kode-konstrukte vir beskou en navigasie van data  Min gebruik van kode om te filter/sorteer/soek | Heeltemal onvoldoende of werk nie korrek nie | **4** |  |
| **Kompleksiteit van data-navrae** | Uitstekende verwerking/data-transformasie  Uitstekende formatering van berekende velde  Verskeidenheid van komplekse navrae, insluitend die gebruik van verwantskappe tussen tabelle | Goeie verwerking/data-transformasie  Goeie formatering van berekende velde  Verskeidenheid van komplekse navrae | Voldoende verwerking/data-transformasie  Beperkte formatering van berekende velde  Min komplekse navrae | Beperkte verwerking/data-transformasie  Swak formatering van berekende velde  Geen komplekse navrae | Heeltemal onvoldoende of nie gedoen nie | **4** |  |
| **Data-instand-houding –** invoeg/bywerk/ skrap van rekords | Gebruik alle data-instand-houdingsaksies  Alle aksies korrek gebruik  Alle toepassing dra by tot ’n funksionele stelsel | Gebruik alle data-instand-houdingsaksies  Alle aksies korrek gebruik  Sommige toepassing dra nie by tot ’n funksionele stelsel nie | Gebruik alle data-instand-houdingsaksies  Alle aksies korrek gebruik  Meeste toepassing dra nie by tot ’n funksionele stelsel nie | Gebruik sommige data-instand-houdingsaksies  Alle aksies nie korrek gebruik  Toepassing dra nie by tot ’n funksionele stelsel nie | Heeltemal onvoldoende of nie gedoen nie | **4** |  |
| **AFVOER** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Korrektheid van afvoer** | Geen logiese foute.  Alle afvoerresultate is korrek | Een geringe logiese fout  Een resultaat problematies | Sommige logiese foute  Sommige resultate nie korrek | Baie logiese foute  Baie resultate nie korrek nie | Afvoer meestal nie korrek nie | **4** |  |
| **Afvoer-komponente en formaat** | Alle afvoer gebruik toepaslik-ste afvoerkomponente, goed geformateer/leesbaar/ge-struktureerd/verstaanbaar | Meeste afvoer gebruik toepaslikste afvoer-komponente, goed geformateer/leesbaar/ge-struktureerd/verstaanbaar | Beperkte gebruik van toepaslike afvoer-komponente met swak formatering | Afvoer meestal nie toepaslik en swak geformateer | Geen afvoer of nie geformateer nie | **4** |  |
| **DOKUMENTASIE** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Kommentaar / Notas**  (Verduideliking van program en kode) | Kode het duidelike notas om alle dele te verduidelik.  Verduideliking toon uitstekende insig.  Uitgebreide programnotas en van 'n uitstekende standaard.  Verduidelik duidelik hoe die program werk | Kode het duidelike notas om alle dele te verduidelik.  Verduideliking toon goeie insig.  Projeknotas is gebruik en is van baie goeie kwaliteit | Kode het duidelike notas om meeste nodigste dele te verduidelik.  Verduideliking toon 'n mate van insig.  Projeknotas is gebruik en is van gemiddelde kwaliteit. | Kode het notas om sekere dele te verduidelik.  Verduideliking toon min insig.  Onvoldoende projeknotas is gebruik | Geen kommentaar of projeknotas | **4** |  |
| **ALGEHEEL** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Voldoen die program aan die vereistes?** | Oorskry vereistes in Fase 1 gestel.  Omvattende program  Alle elemente funksioneer soos gespesifiseer.  Toon insig in alle aspekte | Voldoen aan die vereistes soos gestel in Fase 1.  Minder omvattend  Alle elemente funksioneer soos gespesifiseer.  Toon insig in meeste aspekte | Voldoen aan meeste van die vereistes, maar sommige funksioneer nie goed nie  Slegs sommige program-elemente funksioneer soos gespesifiseer in Fase 1.  Toon insig in 1 of 2 aspekte | Voldoen slegs aan sommige vereistes  Basiese program  Basiese reikwydte  Baie beperkte insig | Voldoen nie aan minimum vereistes nie  Minder as basies  Beperkte reikwydte | **4** |  |
| **Totaal (implementering):** | | | | | | **60** |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Algemeen – Finale produk en indruk Naam van leerder:** | | | | | | | |
| **Aspek** | **4** | **3** | **2** | **1** | **0** |  |  |
| **Vloei van ontwikkeling** | Elke ontwikkelingsfase vloei logies vanaf vorige fase.  Oorspronklike doelwit bereik en voldoen aan al die vereistes gestel in Fase 1 | Voldoen aan ten minste 80% van die vereistes  Sommige aspekte oorspronklik beplan die bereik nie | Voldoen aan meer as 50% van die vereistes  Sommige aspekte moes verander word of op- of afgeskaal word | Meer as 50% van die aanvanklike vereistes nie bereik nie.  Baie aspekte moes verander word of op- of afgeskaal word | Byna geen van die oorspronklike vereistes nagekom nie | **4** |  |
| **Professionele produk** | Bruikbaar en kan in werklike lewe geïmplementeer word.  Baie goed ontwerp en gebruikers-vriendelik  Bevat geen foute nie | Kan met geringe aanpassings in die werklike lewe gebruik word  Goed ontwerp en gebruikers-vriendelik  Bevat minimale foute | Kan met beduidende aanpassings in die werklike lewe gebruik word  Goed ontwerp en gebruikers-vriendelik  Bevat heelwat foute | Nie gereed om in werklike lewe te implementeer nie, maar het potensiaal | Nie gereed om in werklike lewe te implementeer nie  Swak ontwerp | **4** |  |
| **Volledigheid** | Alle fases was volledig, goed ontwerp en uitgevoer.  Alle stadiums en fases goed gedokumenteer. | Alle fases volledig, goed ontwerp en uitgevoer  Alle stadiums en fases goed gedokumenteer, maar geringe tekortkominge | Twee fases was volledig en goed ontwerp en uitgevoer.  Twee fases goed gedokumenteer | Een fase was volledig en goed ontwerp en uitgevoer.  Een fase goed gedokumenteer | Geen fases volledig of goed ontwerp en uitgevoer of goed gedokumenteer nie | **4** |  |
| **Kreatiwiteit** | Produk toon baie kreatiwiteit en oorspronklikheid | Produk toon ’n mate van kreatiwiteit en oorspronklikheid | Produk toon beperkte kreatiwiteit en/of oorspronklikheid | Sekere dele van die produk toon mate van kreatiwiteit en/of oorspronklikheid | Totale gebrek aan kreatiwiteit en oorspronklikheid | **4** |  |
| **Vermoë om kode te verduidelik** | Duidelike en selfversekerde verduideliking van alle geselekteerde kode  Toon uitstekende insig | Verduideliking van alle geselekteerde kode met geringe tekortkominge  Toon insig | Nie in staat om sommige van die geselekteerde kode voldoende te verduidelik nie  Toon mate van insig | Nie in staat om meeste van die geselekteerde kode te verduidelik nie  Beperkte insig | Nie in staat om enige van die geselekteerde kode te verduidelik nie  Geen insig | **4** |  |
| **Houding en toe-wyding** |  |  | Gehou by spertye.  Goed-ontwerpte fases.  Toon uitsonderlike toewyding en trots | Sommige fases was nie betyds nie  Sommige fases toon swak ontwerp  Toon mate van toewyding en trots | Geen fases betyds nie.  Toon geen toewyding nie. | **2** |  |
| **Totaal:** | | | | | | **22** |  |

**Assesseringsopsomming**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fase** | **Fokus** | **Maksimum Punt** | **Punt Toegeken** |
| Fase 1 | Ontleding | 24 |  |
| Fase 2 | Ontwerp | 44 |  |
| Fase 3 | Kodering en Implementering | 60 |  |
| Algemeen |  | 22 |  |
| **Totaal** | | **150** |  |
| **Finalepunt** | |  | |

**Egtheidsverklaring**

|  |
| --- |
| Ek verklaar hiermee dat die werk wat geassesseer is alleenlik die werk van die betrokke leerder is (behalwe waar daar duidelike erkenning en rekord is van enige substantiewe raad/bystand wat aan die leerder gebied is) en dat dit onder toesig/beheerde omstandighede gedoen is om seker te maak dat geen plagiaat gepleeg is nie of die werk nie van iemand anders se werk gekopieer is nie of dat dit voorheen deur enige ander persoon ingedien is nie.  **Kommentaar:**      **Onderwyser se naam: Onderwyser handtekening :Datum:** |

# Bylaag A – Uitleg van Verslag

Gebruik woordverwerkingstyle om jou te help om ’n professionele verslag te skep. Verwys na Kolom 1 hieronder vir voorstelle oor formatering.(Jy kan die inhoudsopgawe outomaties genereer indien jy die formatering gebruik soos in Kolom 1 beskryf (met behulp van style).

|  |  |
| --- | --- |
| TITELBLAD  <title> | Titel van verslag  Jou naam en graad  Inhandigingsdatum |
| **OPSOMMING**  <heading 1> | Oorsig van die verslag  Dui noodsaaklike inligting en aanbevelings aan |
| **INHOUDSOPGAWE**  <table of contents> | Lys van genommerde afdelings in die verslag en hul bladsynommers |
| **INLEIDING**  <heading1> | Verwysingsterme  Die scenario en  oorsig van die verslag |
| **LIGGAAM**  **Opskrifte**  <heading1>  **Subopskrifte**  <heading2> | Opskrifte en subopskrifte wat die inhoud van elke afdeling verteenwoordig  Sluit inligting in oor belangrike idees omtrent die onderwerp  Bespreking van programme wat verwant is aan die scenario |
| **GEVOLGTREKKING**  <heading1> | Stel die gevolgtrekkings wat gemaak kan word uit die inligting wat gevind is  Maak aanbevelings betreffende dit wat jy met die projek beoog |
| **VERWYSINGSLYS**  <heading1> | Lys van bronne wat gebruik is tydens die navorsing  Gebruik vereenvoudigde Harvardstyl/APA styl |
| **AANHANGSEL**  <heading1> | Grafika (bv. skermkopieë) en inligting wat jou navorsing ondersteun maar nie noodsaaklik vir jou verduidelikings is nie. |

# Bylaag B – Egtheidsverklaring

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Naam van leerder** |  | **ID-nommer** | |  |
| **Graad** | 11 | **Jaar** | | 2017 |
| **Vak** | Inligtingstegnologie | | | |
| Praktiese Assesseringstaak (PAT) | | **Onderwyser** | |  |
| Ek verklaar hiermee dat die inhoud van hierdie assesseringstaak my eie oorspronklike werk is (behalwe waar daar duidelike erkenning en toepaslike verwysing na die werk van ander is) en dat dit nie onwettig (deur plagiaat) bekom is, van iemand anders gekopieer is, of voorheen vir assessering deur enige persoon ingedien is nie.  **Lys van hulp ontvang:** | | | | |
| Aard van hulp | | | Persoon wat hulp verleen het | |
|  | | |  | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_ / \_\_\_ / 2017  Handtekening van Leerder Datum | | | | |

# Bylaag C – Terminologie

|  |  |  |  |
| --- | --- | --- | --- |
| **Term** | **Wat is dit** | **Wat dit doen** | **Hoekom dit nodig is** |
| **Taakbeskrywing**  **(Scenario)** | ’n Kort beskrywing, in die leerder se eie woorde, wat die intensie van die taak/projek (PAT) beskryf. Beskryf wat die leerder moet doen sodat die program aan die vereistes van die PAT-spesifikasie voldoen | Definieer die taak vir die leerder, verduidelik wat gedoen moet word.  (Enkele paragraaf) | Om duidelikheid te kry oor wat met die spesifikasies verwag word  Stap 1 in probleemoplossing “Verstaan die probleem”. |
| **Gebruiker** | Die teikengehoor, gebruiker van die program, speler van die speletjie, die leerder in die geval van ’n simulasie, ens. | Voorsien insig in die ontwerpvereistes betreffende gebruiker se kennis, ouderdom, rekenaarvaardighede, geloof, kultuur, taal, geslag, ens. | Om die vlak van die gebruikers se vaardighede en kennis te bepaal vir leiding met ontwerpbesluite |
| **Gebruikerstorie** | ’n Kort storie, wat in ’n sin of twee, waarmee die gebruiker, in alledaagse taal, vertel wat hy/sy met die program wil doen.  Die onderliggende werklike probleem wat die program/stelsel moet oplos.  (Gewoonlik deur die voorgestelde gebruiker geskryf, maar vir praktiese doeleindes, in die geval van die PAT, deur die leerder). | Vertel aan die ontwerper/programmeerder wat die gebruiker wil hê.  Dit definieer watter funksionaliteit in die stelsel ingebou moet word. Spesifiseer WAT nodig is (nie HOE nie)  **Voorbeeld:**  **As ’n**… (***Wie*** of akteur of gebruiker)  **wil ek** …( ***Watter*** vermoë of kenmerk benodig word)  **sodat**… (***Hoekom*** is dit van waarde of tot voordeel) | Om vereistes kenmerk-vir-kenmerk te spesifiseer. (funksie-vir-funksie)  Om uit te redeneer wat die program/stelsel moet voorsien  Om te verseker dat vereistes in klein, hanteerbare funksionele stukkies afgebreek word, m.a.w. individuele kenmerke wat as ’n *enkele taak* implementeer kan word. |
| **Gebruiksgeval (Use Case)**  **(**stelsel funksie/ kenmerk) | ’n Gebruiksgeval beskryf hoe ’n tipe gebruiker (ook genoem ’n akteur) die program/stelsel gebruik om ’n spesifieke doelwitte bereik.  Elke gebruiksgeval dui op baie scenario’s | ’n Gebruiksgeval beskryf *funksionele* vereistes vanuit die gebruiker se oogpunt.  ’n Gebruiksgeval verteenwoordig   * ’n doel van ’n akteur (tipe gebruiker) met die gebruik van die stelsel, bv.*Onttrek kontant (*rede vir gebruik vanstelsel); * scenario’s (uitvooerbaar), ofvolgorde van stappe/ aksies/gebeurtenissewat uitgevoer word, verteenwoordig verskillende deur ’n gebruiksgeval in bereiking van die doelwit,soos: {Spesifiseer Rekening, Spesifiseer bedrag, Neem kontant} (vir*Onttrek kontant*). * die stelsel betrokke/wat gebruik word, bvg. OTM stelsel | Om ’n spesifieke gebruik (doelwit/kenmerk/ funksie) van die program/stelsel deur ’n akteur (gebruiker van stelsel) betreffende die stappe/ aksies wat die gebruiker sal uitvoer om die doelwit te bereik, te spesifiseer  Om as basis te dien om toetsgevalle (test cases) te ontwikkel meestal op gebruikers-aanvaardingsvlak  Om die vloei van prosesse in ’n stelsel te beskryf  Genereer toetsgevalle/voorsien prototipe toetse |
| **Gebruiksgeval-diagram**  **(Use Case Diagram (UCD))** | ’n UCD is ’n grafiese voorstelling van ‘n gebruiks-geval  **Let Wel:** It does NOT represent ALL the elements of the use case, such as the description of the sequence of steps/actions/events that must be performed or the exceptions/ alternative flows (scenarios) – this could be done using additional, separate UCDs | Verteenwoordig die funksionaliteit (gebruik) van die stelsel, m.a.w. dit wys die gebruiker se doelwitte  Grafiese beskrywing van   * akteurs (gebruikers) * gebruiksgevalle (*use cases*) * stelselgrense, en * die verwantskap tussen alles | Om die gebruike, (gewoonlik op hoë/topvlak)van die stelsel en die verwantskap tussen die gebruiker en die stelsel te toon  Om   * die konteks van die stelsel te spesifiseer * Vereistes vas te lê * Implementering te bestuur |
| **Akteur**  **(gebruiker)** | Iemand (of iets) wat die stelsel/ stelselkenmerk/funksie gebruik, bv. ’n persoon, toestel, eksterne programkomponent, ander stelsel, sensor, tydhouer, ens.  (’n Akteur is ’n tipe gebruiker van ’n stelsel) | Akteurs gebruik die stelsel op ’n interaktiewe wyse deur op knoppies te klik, teks in hokkies te tik, op ikone te klik, bv. eindgebruiker, administrateur, tydhouer, ens. om ’n doelwit te bereik  ’n Akteur   * aktiveer ’n gebruiksgeval (funksie/kenmerk van die stelsel) * het verantwoordelikheid teenoor die stelsel (toevoer) * het verwagtinge van die stelsel (afvoer) | Akteurs moet   * ekstern tot die stelsel wees * Dien as bron en bestemming vir data (eksterne objekte wat data voorsien/gebruik) |
| **Scenario** | ’n Scenario is  ’n uitvoerbare reeks gebeure/stappe/aksies om ’n doelwit te bereik  Bv. {Kies rekening, Spesifiseer Bedrag, Neem kontant} (vir die *Onttrek kontant* gebruiksgeval)  ’n Gebruiksgeval bestaan uit verskeie scenario’s:  Hoof scenario – beskryf die noodsaaklike (korste pad) na sukses (bereiking van doelwit) – elke stap is noodsaaklik (kan nie oorgeslaan word nie) en elke stap is suksesvol.  Addisionele scenario’s – ander paaie/alternatiewe stappe na sukses, sommige wat tydelik misluk, dan herstel, maar eindig in sukses (Alternatiewes), ander wat misluk (Uitsonderings (*Exceptions*)) | Beskryf die vloei van gebeure/aksies/stappe vanaf die oomblik wat die akteur die gebruiksgeval aktiveer/inisieer totdat die doelwit bereik is:   * Hoe en wanneer die kenmerk geaktiveer/begin word * Wisselwerking/interaksie tussen die stelsel en die akteur en watter data hulle uitruil * Wanneer die gebruiksgeval data wat in die stelsel gestoor is, gebruik of data in die stelsel stoor * Hoe en wanneer die gebruiksgeval eindig   Elke scenario in ’n gebruiksgeval lei tot die formulering van een of meer toetsgevalle(*test cases*) (scenario dui op toetsgevalle) | Om die stappe/aksies te beskryf wat ’n gebruiker moet uitvoer om ’n doelwit te bereik.  Om alternatiewe scenario’s te identifiseer, vra: “Wat kan verkeerd gaan?”, bv.   * Verkeerde toevoer deur akteur? (bv. as akteur ongeldige PIN intik) * Watter besigheidsreëls geld? (bv., die akteur spesifiseer meer geld as wat in sy rekening beskikbaar is) * Wat kan verkeerd gaan? (bv. kaart het verval) |
| **Toetsgeval** | ’n Toetsgeval is ’n stel   * toetstoevoer, bv. toetsdata * uitvoervoorwaardes (aksies/ gebeurtenisse/toetse wat uitgevoer word) * verwagte resultate   wat vir ’n spesifieke doelwit (gebruiksgeval)/spesifieke aspek/kenmerk ontwikkel is, soos om ’n spesifieke programroete uit te voer of om die voldoening aan ’n spesifieke vereiste te verifieer | Dit help die toetser/programmeerder om foute/swakhede/moontlike faling uit te wys en dit reg te maak  Ondersoek toevoere en afvoere om te bepaal of ’n stelsel/eenheid korrek werk  Elke vereiste of doelwit wat van die program verwag word om te bereik, het ten minste een toetsgeval nodig  Voorbeeld van toetsgevalle vir suksesvolle onttrekking van kontant:   1. Verifieer die bedrag ingesleutel 2. Verifieer die rekening se balans 3. Verifieer die daaglikse limiet 4. Verifieer die hoeveelheid geld beskikbaar in die OTM | Om mislukkings (faling) op te spoor of voldoening aan vereistes te verifieer  Om foute uit te wys, m.a.w. om funksionaliteit te toets  Om te verifieer dat die program aan die gebruiker se behoeftes te voldoen  Om die programmeerder te verseker dat die program doen wat verwag word dit moet doen |
| **Aanvaardingstoets**  **(Bevestigings)** | Aanvaardingstoetse is  toetsgevalle wat vanuit gebruikerstories/gebruiks-gevalle ontwikkel word en verteenwoordig ’n verwagte resultaat vanaf die stelsel (bereiking van die doel/waarde wat die gebruiker uit die stelsel sal kry, bv. die kontant).  Uiteindelik voorsien dit die kriteria waarteen die uitkoms of doelwit van die gebruikerstorie/vereistes getoets kan word. | Verifieer dat die doelwit van die gebruikerstorie/gebruiksgeval bereik is.  Sê vir die gebruiker hoe die doelwit/funksionaliteit bevestig gaan word.  Sê aan die ontwerper/programmeerder hoe hy/sy sal weet dat ’n gebruikerstorie/gebruiksgeval korrek geïmplementeer is.  Verseker dat elke program uitvoer, alhoewel slegs met die geïmplementeerde funksies.  **Ek**(akteur) **weet dit is suksesvol/bereik wanneer**  (akteur) bv. Ek  (werkwoord/aksie) bv. sien, doen, hoor, neem, ens.  (Waarneembare resultaat) bv.(sien) boodskap, (neem) kontant, ens. | Sodat die programmeerder sal weet wanneer, dit wat die gebruiker wou hê; bereik is.  Sodat die gebruiker sal weet wanneer die taak/eenheid voltooi is en as voltooid afgemerk kan word.  Om te verseker die program is ontwerp om die gebruiker se kriteria te slaag.  Help om scenario’s te identifiseer waaraan gebruikers/ontleders en/of ontwerpers moontlik nie gedink het nie (identifiseer onvolledige gebruikerstories of *spikes*). |
| **Eenheidstoets** | Eenheidstoets is  toetsgevalle wat deur die programmeerder ontwikkel word om ’n funksionaliteit/een kenmerk op ’n slag te toets.  (’n Eenheid is die kleinste toetsbare deel van ’n program/stelsel, soos funksies/prosedures, metodes, koppelvlakke, ens.) | Wys programmeringsfoute uit | Om elke gedeelte van ’n program te isoleer en te wys dat al die individuele dele korrek is.  Om te verseker dat die kode aan die ontwerp voldoen en uitvoer soos dit veronderstel is om uit te voer. |

# Bylaag D – Voorbeeld

**OTM-Stelsel - Voorbeelde**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Gebruiker-behoeftes** | **Gebruikerstories** | | | | | **Aanvaardingstoets** | | | | |
|  | As ‘n kaarthouer wil ek geld trek sodat ek my rekeninge kan betaal  As ‘n kaarthouer wil ek my rekeningbalans sien sodat ek kan weet hoeveel geld ek beskikbaar het  As ‘n kaarthouer wil ek inteken sodat ek ’n transaksie kan doen  As ‘n kaarthouer wil ek geld oorplaas sodat ek ’n ander rekening kan optop  As ‘n bankamptenaar wil ek die OTM deaktiveer sodat ek banknote daarin kan plaas  As ‘n bankamptenaar wil ek daaglikse OTM transaksies druk sodat ek sekere verifiëring kan doen | | | | | Ek sal weet dit is gedoen wanneer ek die kontant neem  Ek sal weet dit is gedoen wanneer ek my rekeningbalans sien  Ek sal weet dit is gedoen wanneer ek die transaksie menu sien  Ek sal weet dit is gedoen wanneer ek ’n kwitansie met die nuwe balans ontvang  Ek sal weet dit is gedoen wanneer ek die boodskap sien dat OTM sluit  Ek sal weet dit is gedoen wanneer ek die drukstuk met transaksies ontvang | | | | |
| **Doelwitte** | Begin Sessie (Log in), Onttrek kontant, Beskou balans, Dra fondse oor, Deaktiveer OTM, Druk transaksiegeskiedenis | | | | | | | | | |
| **Gebruiksgeval Diagram**  (top-vlak dienste wat die stelsel aan sy akteurs voorsien) |  | | | | | | ’n Topvlak UCD is ’n volledige beskrywing van die stelsel se funksionaliteit/dienste, alhoewel besonderhede/detail mag kort.  ’n Topvlak-funksionaliteit/diens moet so wees dat die akteur, in ’n enkele sessie, slegs die spesifieke funksie/diens kan versoek/uitvoer. | | | |
| **Scenario’s**  (vir *Begin Sessie*) | **Hoof scenario:** (alles word reg gedoen) | | **Alternatiewe scenario:** (foute kom voor, bv. ongeldige toevoer) | | | | | **Uitsondering:**(mislukking) | | |
| **Gebruiker** | **Stelsel** | **Gebruiker** | | **Stelsel** | | | **Gebruiker** | | **Stelsel** |
|  | Sit kaart in  Sleutel PIN in | Valideer aard, vra vir PIN  Valideer PIN Valideer rekeningnommer Laat toegang toe/Vertoon menu | Sleutel foutiewe PIN in | | Vra om weer in te sleutel (Slegs 2X) | | | Foutiewe PIN 3X verkeerd ingesleutel | | Sluk kaart Vertoon boodskap |
| **Toetsgevalle**  (vir *Onttrek kontant*) | **Moontlike toetsgevalle** | | | **Toevoerdata** | | | | | **Verwagte resultaat** | |
| Verifieer rekening balans | | | Bedrag<= Balans | | | | | Sukses | |
|  | | | Bedrag> Balans | | | | | Waarskuwingsboodskap  Nog ’n kans om bedrag in te sleutel | |
| Verifieer daaglikse limiet | | | Bedrag oorskry nie daaglikse limiet nie | | | | | Sukses | |
|  | | | Bedrag oorskry daaglikse limiet | | | | | Waarskuwingsboodskap  Nog ’n kans om bedrag in te sleutel | |
| Verifieer hoeveelheid geld in OTM | | | Geld beskikbaar>= Bedrag | | | | | Sukses | |
|  | | | Geld beskikbaar<Bedrag | | | | | Boodskap  Skop kaart uit  OTM sluit | |

# Riglyne vir onderwysers om die leerders leiding te gee

## Wat word daar van die leerders verwag om te doen en te lewer?

Daar word van die leerder verwag om onder toepaslike toesig die volgende te doen:

* Kies 'n area van belangstelling binne die gegewe onderwerp/scenario te kies
* Identifiseer die fokusarea wat ondersoek/nagevors moet word
* Beplan, doen navorsing en voer die projek uit
* Skryf 'n verslag vir 'n spesifieke gehoor
* Ontwikkel 'n sagtewareoplossing volgens die vereistes
* Lewer bewys van al die fases van die projek vir assessering

## Hoe sal die leerders te werk gaan?

Die leerder sal:

* 'n Individuele projek beplan en voltooi deur 'n verskeidenheid programmering- en sagtewareingenieursvaardighede en -strategieë toe te pas om die doelwitte, soos in die PAT-vereistes uiteengesit is, te bereik
* Vrae identifiseer om te vra
* Inligting uit 'n verskeidenheid bronne bekom, krities uitsoek en gebruik, data verwerk en analiseer, dit binne verband toepas en insig toon met sinvolle aansluiting, verband en kompleksiteit ten opsigte van die onderwerp en die vraag waarop gefokus word
* 'n Verskeidenheid vaardighede kies en gebruik, insluitend ontwerphulpmiddels en algoritmes, probleme oplos, krities, kreatief en aanpasbaar besluite neem, om 'n sagteware-oplossing te lewer
* Uitkomste evalueer, beide in verband met PAT-vereistes en eie leer en werkverrigting
* Toepaslike kommunikasievaardighede en media gebruik om bewyse in 'n gepaste formaat aan te bied

## Vaardighede wat vereis word

Die leerder moet in staat wees om:

* Navorsing oor die onderwerp te doen en bevindinge behoorlik te dokumenteer, insluitend verwysings soos gespesifiseer in Fase 1 se nasienblad
* 'n Volledige analise van gebruikervereistes te doen wat 'n volledige beskrywing van die rol, aktiwiteite, vereistes en beperkinge van ten minste TWEE verskillende gebruikers van die beplande stelsel insluit
* Inligting bymekaar sit om by die inhoud en doel te pas
* Besluitneming- en probleemoplossingsvaardighede toe te pas
* Beplanning-, navorsing-, kritiese denke-, analise-, sintese-, evaluering- en aanbiedingsvaardighede uit te brei
* Selfvertroue te ontwikkel in die toepassing van inhoud, programmering en sagteware ingenieurswese-beginsels en -tegnieke wat hulle bestudeer het
* Vaardighede skeppend te ontwikkel en toe te pas deur inisiatief en ondernemingsgees te toon
* Raad en ondersteuning te vra, wanneer nodig

## Wat moet die leerder vooraf geleer word?

Voordat daar met die PAT begin word, moet die leerder die volgende geleer word:

* Programmeringskode, -beginsels en -tegnieke
* Sagteware-ingenieurswesebeginsels en -tegnieke, insluitend die gebruik van sagtewareontwerp
* Projekbestuurvaardighede, insluitend tyd-, hulpbron- en taakbestuur
* Die formaat en struktuur van aanvaarde vorme van navorsingsverslae, insluitend 'n uittreksel, inleiding, bespreking met verwysings na alle bronne, gevolgtrekking, verwysings.

## Wanpraktyk

Leerders mag NIE:

* Hulp/leiding van ander kry sonder om erkenning aan hulle te gee nie (voltooi Bylae B vir ELKE fase)
* Werk inlewer wat nie hulle eie is nie, bv. Programkode wat deur iemand anders geskryf is
* Hulle PAT-werk aan ander leerders leen nie
* Ander leerders toelaat om toegang te kry tot, of gebruik te maak van, hulle eie werk nie (dit beteken nie dat kandidate nie hulle boeke vir ander kandidate mag leen nie, maar kandidate mag nie plagiaat met ander leerders se werk te pleeg nie)
* Werk insluit wat direk uit boeke, vanaf die Internet of uit ander bronne gekopieer is, sonder om erkenning en dank te gee nie (dit mag nie 20% van die werk wat ingelewer word, oorskry nie)

Hierdie handelinge kom neer op wanpraktyk, waarvoor straf toegepas sal word.

Indien wanpraktyk geïdentifiseer is, moet die assesseringsgesag in kennis gestel word en besonderhede van enige werk wat nie die leerder se eie is nie, moet aangeteken word.

## Leerder se verklaring van egtheid van die PAT

Vir elke fase voltooi leerders 'n verklaring (Bylae B) vir die werk wat tydens daardie spesifieke fase gedoen is. Alle toepaslike raad/hulp wat aan die leerders gegee is, moet as deel van die fase se dokumentasie aangeteken word. .

## Rol van die onderwyser

Die onderwyser sal die inligtingsbestuur-inhoud, vaardighede en strategieë onderrig voor die aanvang van die projek.

Die onderwyser sal die projek en leerders bestuur en toesig hou deur:

* 'n Aanvanklike beplanningsoorsig te doen om die onderwerp/scenario, vereistes, doelstellings en ontwikkeling van die projek te bespreek.
* Vooraf lees te fasiliteer om agtergrondinligting te verkry oor die onderwerp/scenario.
* Gereelde terugvoer aan leerders te gee, bv. oor hulle navorsing en analise.
* Die leerders se werk aan die einde van elke fase te assesseer, deur van die gestandaardiseerde assesseringsinstrument gebruik te maak.
* Die leerder se assessering vir elke fase onderskryf deur die assesseringsinstrument te onderteken, insluitend die finale verklaring dat die werk wat die leerder ingehandig het vir assessering die leerder se eie is sonder enige hulp.
* Die algemene assessering te bevestig deur gebruik te maak van deurlopende waarneming en terugvoering om 'n finale indruk te verkry aangaande onafhanklike werk, beplanning, insig en probleemoplossing.
* Assessering te doen van leerders se werk deur enige standaardisering- en interne modereringsprosedures te volg wat vereis word.
* 'n Ondervragingsessie te hou ten einde vas te stel wat die leerder se begrip en insig van sy/haar projek is.

Die onderwyser sal die potensiële projek (analise, ontwerp) assesseer ten opsigte van die volgende kontrolelys.

* Is die fokusarea geskik vir die projek?
* Laat die fokus die leerder toe om hoërorde-konsepte en -vaardighede in die assesserings doelstellings te ondersoek en toegang te verkry daartoe, bv. navorsing, analise, beplanning, evaluasie, verduidelik en integreer eerder as blote beskrywing en vertelling?
* Is die fokusarea en voorgestelde aksie duidelik en gefokus op 'n kwessie wat binne die tydraamwerk bestuur kan word met die beskikbare tyd en hulpbronne?
* Dui die fokus en voorgestelde aksie aan dat die leerder in staat sal wees om die onderwerp te ondersoek en na te vors, en om die aktiwiteit of taak onafhanklik uit te voer binne toepaslike etiese en metodologiese riglyne?
* Is dit waarskynlik dat die leerder probleme sal ondervind om die taak en kwessies wat met die fokus area geassosieer word, te verstaan?

Die onderwyser sal die PAT sertifiseer

* Die onderwyser moet op die assesseringsinstrument bevestig dat die werk wat assesseer is slegs dié van die betrokke leerder is en dat dit uitgevoer is onder beheerde toestande met toesighouding.
* Onderwyser moet die assesseringsinstrument vir elke fase teken.

## Toesig/Gekontroleerde toestande

Die PAT moet op so 'n wyse bestuur word dat dit moontlik is om te bevestig dat die werk wat assesseer word alleenlik die werk van die betrokke leerder is.

## Bestuur van die PAT

Die onderwyser moet sy/haar werkskedule beplan volgens die tyd wat toegeken is aan die PAT in die KABV-dokument vir Inligtingstegnologie (onderrigplan vir Graad 11).

Daar is verskillende moontlike benaderings ten opsigte van die bestuur van die PAT:

Opsie 1:

* Die onderwyser kan 'n deel van die tyd op 'n weeklikse basis aan die PAT wy, terwyl daar terselfdertyd met normale onderrig van die Graad 11-kurrikulum voortgegaan word in die res van die week.
* Indien hy/sy die opsie kies, moet daar aan die einde van die eerste kwartaal met die PAT proses begin word.

Opsie 2:

* Die onderwyser kan 'n aaneenlopende tydperk aan die PAT wy, bv. die weke aan die einde van die tweede kwartaal en die begin van die derde kwartaal.

Dit word aanbeveel dat die onderwyser leerders se onderwerpe/fokusareas aanteken wanneer hulle met Fase 1 begin, om 'kitsprojekte' te voorkom wat waarskynlik nie die leerder se eie werk sal wees nie.

## Bewyse van assessering

Bewyse wat aangebied word vir assessering moet toon hoe die leerder die assesseringsdoelwitte en -kriteria bereik het en moet beplanning, terugvoer en vordering met projek insluit.

Bewyse vir assessering sal die volgende insluit:

* Die projekproduk, insluitend die dokumentasie wat vir elke fase ingedien word.
* Die voltooide leerder-assesseringsinstrument (vir elke fase)

## Vereistes

**(Nasionale Protokol vir Assessering Grade R – 12, Hoofstuk 3)**

Praktiese Assesseringstaak-komponente moet:

* Uit assesseringstake bestaan wat die leerder se PAT-punt verteenwoordig soos beoog in Hoofstuk 4 van die Kurrikulum- en Assesseringsbeleidstelling vir IT.
* 'n Punt insluit wat toegeken is vir elke assesseringstaak (fase) asook 'n gekonsolideerde punt
* 'n Riglyn wees vir die assesseringskomponente soos gespesifiseer in Hoofstuk 4 van die Kurrikulum- en Assesseringsbeleidstelling vir IT.
* Beskikbaar wees vir monitering en moderering.
* Geëvalueer word, gekontroleer word en gesertifiseer word deur die onderwyser voordat dit voorgelê word as bewys van die leerder se prestasie.

## Versuim

**(Nasionale Protokol vir Assessering Grade R – 12, Hoofstuk 3)**

Die afwesigheid van 'n PAT-punt vir IT, sonder 'n geldige rede, sal daartoe lei dat 'n leerder nie resultate vir die vak ontvang nie.

Die leerder sal tot drie weke voor die aanvang van die jaareindeksamen gegun word om uitstaande werk in te handig of aan te meld vir die PAT. Sou die leerder versuim om die uitstaande PAT-vereistes te vervul, sal aan so 'n leerder nul (“0”) toegeken word vir die PAT-komponente wat nie voltooi/ingehandig is nie.

In die geval dat 'n leerder nie voldoen aan die vereistes van die PAT nie, maar waar 'n geldige rede gegee word:

* Mag hy/sy nog 'n geleentheid gegun word om geassesseer te word in die opgelegde take, afhangend van die besluit van die Hoof van die assesseringsliggaam.
* Die leerder moet, binne drie weke voor die aanvang van die jaareindeksamen uitstaande werk inhandig of aanmeld vir die PAT.
* Sou die leerder versuim om die uitstaande PAT-vereistes te vervul, sal die punte vir hierdie PAT-komponente weggelaat word en die finale punt aangepas word vir promosiedoeleindes in terme van take wat voltooi is.

LET WEL:

In die geval van IT is so 'n situasie nie baie waarskynlik nie (tensy 'n leerder se siekte oor 'n lang tydperk sou strek) aangesien die PAT nie 'n eenmalige eksamen/taak is nie.

* Geldige redes in hierdie konteks sluit die volgende in:
* Siekte, bevestig deur 'n geldige mediese sertifikaat, uitgereik deur 'n geregistreerde mediese praktisyn
* Menslikheidsredes, wat insluit dood van 'n nabye familielid, bevestig deur 'n doodsertifikaat
* Wanneer die leerder in 'n hofsitting moet verskyn, wat bevestig word deur geskrewe bewyse
* Enige ander rede wat deur die Hoof van die assesseringsliggaam, of deur sy of haar gevolmagtigde, as geldig aanvaar mag word

In die geval dat 'n leerder sou versuim om te voldoen aan die vereistes van die Praktiese Assesseringstaak-vereistes van 'n spesifieke vak, en waar geldige redes verskaf word, moet die bewyse van sodanige geldige redes ingesluit word by die bewyse van leerderprestasie.